CLAIMS:

1. A method of recovering a clock signal for a TDM output from packets of TDM data which have been transmitted over a packet network, the method comprising:

providing a packet buffer to store incoming packets after transmission over the packet network,

maintaining a packet count which is incremented as packets arrive at the packet buffer, and decremented each time a packet leaves the packet buffer, and

sampling the packet count and controlling the clock frequency of the TDM output on the basis of the sampled packet count.

2. A method as claimed in claim 1, which further comprises:

sampling the packet count at a fixed interval,

performing a calculation to determine the source frequency of a TDM clock at the source of the packets, and

writing a new local frequency value to a digitally controlled oscillator which controls the clock frequency of said TDM output.

3. A method as claimed in claim 1, which further comprises filtering the value of the packet count before sampling the packet count.

4. A method as claimed in claim 3 wherein the filtering is carried out using a first order low pass filter.

5. A method as claimed in claim 1 which further comprises:

making adjustments to the packet buffer, by adding or removing packets, based on at least a filtered reading of the depth of the packet buffer.

6. A method as claimed in claim 1, which comprises controlling said clock frequency so as to maintain the average depth of the packet buffer at a predetermined value.

7. A method as claimed in claim 6, which comprises performing the following algorithm to determine said clock frequency:

$$F_m = F_{m-1} + G1(Y_m - Y_{m-1}) + G2(Y_m - Offset)$$

where:

F_m and F_{m-1} are the new and previous clock frequencies respectively;

G1 and G2 are constants;

Y_m and Y_{m-1} are new and previous filtered packet count values; and

Offset is a constant that represents the desired depth of the packet buffer.

8. A reference clock recovery system, for recovering a clock signal for a TDM output from packets of TDM data which have been transmitted over a packet network, the system comprising:

a packet buffer for storing incoming packets after transmission over the packet network,

a packet counter which maintains a packet count which is incremented as packets arrive at the packet buffer, and decremented each time a packet leaves the packet buffer, and

a clock control device which samples the packet count value and controls the clock frequency of the TDM output on the basis of the sampled packet count.

9. A reference clock recovery system as claimed in claim 8, which further comprises a digitally controlled oscillator which controls the clock frequency of said TDM output,

and wherein the clock control device performs a clock control algorithm which determines the source frequency of a TDM clock at the source of the packets, and writes a new local frequency value to the digitally controlled oscillator so as to control the clock frequency of said TDM output.

10. A reference clock recovery system as claimed in claim 8, which further comprises a packet counter filter arranged to filter the value of the packet count before the value of the packet count is sampled by the clock control device.

11. A reference clock recovery system as claimed in claim 8, which further comprises a buffer depth control device arranged to make adjustments to the packet buffer, by adding or removing packets, based on at least a filtered reading of the depth of the packet buffer.

12. A reference clock recovery system as claimed in claim 8, wherein said clock control device is arranged to control said clock frequency so as to maintain the average depth of the packet buffer at a predetermined value.

13. A reference clock recovery system as claimed in claim 12, wherein said clock control device performs the following algorithm in order to control said clock frequency:

$$F_m = F_{m-1} + G1(Y_m - Y_{m-1}) + G2(Y_m - Offset)$$

where:

F_m and F_{m-1} are the new and previous clock frequencies respectively;

G1 and G2 are constants;

Y_m and Y_{m-1} are new and previous filtered packet count values; and

Offset is a constant that represents the desired depth of the packet buffer.